

Application Number : 10/052,585 Confirmation Number: 2571  
Applicant : Mark Webster Newman et al.  
Filed : 23 January 2002  
TC/A.U. : 2111  
Examiner : Vu, Trisha U  
  
Docket Number : PARC-DA1084  
Customer No. : 22,835

Proposed Amendment and Interview Summary

Via Fax <571-273-3643> and Electronic Filing

### **PROPOSED AMENDMENT AND INTERVIEW SUMMARY**

Dear Examiner Vu:

In light of the interview on 12 September, please find the proposed amendment and the interview summary below.

#### **Identification of Claims and Reference Discussed**

Claims for discussion: Independent claims 1, 11, and 20.

References for discussion: Pleso (U.S. Patent No. 6,009,480, hereinafter "Pleso").

#### **Applicant's Arguments**

Applicant respectfully points out that the instant application teaches using a universal contextual interface to provide context information, wherein the universal contextual interface **allows components using different operating systems, communication protocols, file formats, and data types to transfer context information between each other without requiring the devices to use domain-specific interfaces, protocols or data format** (see paragraphs [0008] and [0015] of the instant application).

This is beneficial because it allows arbitrary components (such as computer applications or programs, data, memory, file directories, individual files, printer devices, cellular telephones, facsimile machines, copier machines, scanner devices, desk-top computers, lap-top computers, personal digital assistant (PDA) systems) to successfully communicate with each other without having a priori knowledge of particular domain-specific protocols (such as the file system domain (e.g., NFS and CIFS) or the printer domain (e.g., IPP and LPR)), operation systems, or data format, of each other (see paragraph [0003] of the instant application).

Additionally, the instant application teaches that the universal contextual interface comprises instructions that are particular to the host device, wherein the instructions can be understood and performed by a plurality of other devices to **enable the plurality of other devices to communicate and transfer contextual data with the host device**; provide event notifications to the plurality of devices with respect to changes in contextual data for the host device; **enable the plurality of other devices to receive user interfaces to allow users of the plurality of other devices to view changed contextual data or enable the plurality of devices to receive data from the first device**. More specifically, the contextual data **includes information with respect to the host device including type, owner, history of use, whether the first device is currently in use, other operating status information, identity, location on network** (see paragraphs [0027]-[0028] of the instant application).

Furthermore, the instant application teaches that **the universal contextual interface is implemented in Java** (see paragraph [0033] of the instant application). Note that Java is executable data that can be transmitted to a requesting device where it may be executed.

In contrast, Pleso discloses a serial interface (I/F) 82 for downloading a device driver, (including a most up-to-date peripheral device driver) for a peripheral device directly from the peripheral device to a computer system (see Pleso, col. 7, lines 7-9 and col. 13, lines 25-35). However, Pleso **fails to suggest**,

**either explicit or implicit, that (1)** the serial interface 82 allows devices coupled through the serial interface 82 to use different operating systems, communication protocols, file formats, and data types to transfer context information between each other without requiring the devices to use domain-specific interfaces, protocols or data format; **(2)** the serial interface 82 is implemented in Java; and **(3)** the serial interface 82 includes instructions that can be understood and performed by a computer system to enable the computer system to communicate and transfer contextual data with the peripheral device; enable the computer system to receive user interfaces to allow users of the computer system to view changed contextual data or enable the computer system to receive contextual data from the peripheral device, wherein the contextual data includes information with respect to the peripheral device including type, owner, history of use, whether the peripheral device is currently in use, other operating status information, identity, location on network.

**Proposed Amendment:**

Applicant proposes to amend independent claims 1, 11, and 20 to clarify that the present invention provides a universal contextual interface which allows components using different operating systems, communication protocols, file formats, and data types to transfer context information between each other without requiring the devices to use domain-specific interfaces, protocols or data format. Furthermore, Applicant has amended these independent claims to clarify that this universal contextual interface is implemented in Java. Additionally, Applicant has amended these independent claims to clarify that this universal contextual interface includes instructions that can be understood and performed by a second device to enable the second device to communicate and transfer contextual data with the host device; enable the second device to receive user interfaces to allow users of the second device to view changed contextual data or enable the second device to receive contextual data from the host device, wherein the contextual data includes information with respect to the host device including

type, owner, history of use, whether the peripheral device is currently in use, other operating status information, identity, location on network. These amendments find support in paragraphs [0003], [0008], [0015], [0027]-[0028], and [0033] of the instant application.

1. A system comprising:

a plurality of devices, wherein devices within the plurality of devices communicate with incompatible protocols;

a first device in the plurality of devices having a universal contextual interface, ~~wherein the universal contextual interface is implemented in Java~~~~the universal contextual interface associated with at least one instruction for transferring contextual data associated with the first device; and~~

wherein the universal contextual interface comprises instructions that are particular to the first device, wherein the instructions can:

be understood and performed by the plurality of devices to enable the plurality of devices to communicate and transfer contextual data with the first device;

provide event notifications to the plurality of devices with respect to changes in contextual data for the first device; and

enable the plurality of devices to receive user interfaces to allow users of the plurality of devices to view changed contextual data or enable the plurality of devices to receive data from the first device; and

wherein contextual data includes information with respect to the first device including type, owner, history of use, whether the first device is currently in use, other operating status information, identity, location on network, administrative domain, information with respect to one or more users of the first device or files stored at the first device; and

a second device in the plurality of devices that invokes the universal contextual interface of the first device by executing the instructions at least one

~~instruction to transfer the contextual data associated with the first device between the first device and at least one of the other devices in the plurality of devices, the plurality of devices having no prior knowledge of each other;~~

wherein the universal contextual interface is directly invoked by the second device to allow the contextual data to be transferred to the second device;  
and

wherein the second device registers as a listener with the first device through a notification interface of the first device to receive event notifications with respect to changes in the contextual data associated with the first device; and wherein the universal contextual interface allows components using different operating systems, communication protocols, file formats, and data types to transfer context information between each other without requiring the components to use domain-specific interfaces, protocols, or data format.

### **Outcome of Interview**

In the telephone interview conducted on **12 September 2008**, Applicant and Examiner discussed possible amendment options. Applicant proposed during the interview that the new amendment may include language to further clarify that the universal contextual interface is unique because it: (1) allows components using different communication protocols and data types to transfer context information between each other without requiring the components to use domain-specific interfaces, protocols or data format; and (2) includes mobile code.

Examiner pointed out that the Applicant might consider including languages to further distinguish the universal contextual interface from interface 82 in Pleso. Furthermore, Examiner suggested that the Applicant further clarify that scope of contextual data, so as to distinguish from device drive associated data in Pleso. Examiner also suggested that the Applicant be more specific about mobile code.

Applicant agreed with the Examiner's suggestion changes and subsequently incorporated these changes into the amended claims. The Examiner indicated that further consideration and/or search are needed.

Respectfully submitted,

By / Shun Yao/  
Registration No. 59,242

Date: 10 November 2008

Shun Yao  
PARK, VAUGHAN & FLEMING LLP  
2820 Fifth Street  
Davis, CA 95618-7759  
Tel: (530) 759-1667  
Fax: (530) 759-1665  
Email: shun@parklegal.com